

Описание решения по шардированию диалогов.

Время составления: 27.07.2021

1. Компоненты

1. Шардирование осуществлялось с помощью ProxySQL + MySQL. Конфигурация в каталоге docker.
2. В качестве ключа шардирования использовалось поле shard_id определяется по механизму Consistent hashing, так же есть поле hash_id для целей решардинга. Скрипты миграции в каталоге db/migrations.
3. Основная таблица подлежащая шардингу dialog_message.
4. В тестовой конфигурации использовалось два шарда с номерами 0 - основной и 1.

2. Исправленно по замечаниям на Consistent hashing.

1. Шарды добавляются при помощи механизма Consistent hashing.
2. Для определения соответствия шарда использовался пользователь получатель сообщения. Пользователь получатель сообщения указывается в таблице dialog_message поле to_user.
3. У пользователя получателя определяется город проживания (таблица user поле city), по которому согласно механизма Consistent hashing (так же поле city) определяется номер шарда.

3. Алгоритм для ключа записи

1. Ключ записи в dialog_message определяется функцией RandomIdWithShardId на стороне микро-сервиса backend.
2. Данная функция генерирует id в зависимости от времени плюс случайное число плюс номер шарда (ограничение до 255 шардов).